Homework II

Due date: Feb 25th 2024 at 11:59pm

Submission Guidelines: 1. Put all the documents into one folder, name that folder as firstnamelastname_UIN and compress it into a .zip file.

2. For coding problems, your submission should include a code file (either .py or .ipynb), and also a pdf file (in one file together with other non-coding questions) to report the results that are required in the question.

Problem 1: Least Absolute Deviation (15 points)

In class, we have assumed the following data generative model

$$y = f(\mathbf{x}) + \epsilon$$

where ϵ follows a standard gaussian distribution, i.e., $\epsilon \sim \mathcal{N}(\epsilon|0, 1)$. Assume a linear model for $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ (the bias term is ignored here). We now modify the data generative model by assuming that ϵ follows a Laplacian distribution whose probability density function is

$$p(\epsilon) = \frac{\lambda}{2} \exp(-\lambda|\epsilon|)$$

where λ is a positive constant. For more about Laplacian distribution please check the following wiki page `http://en.wikipedia.org/wiki/Laplace_distribution`.

Based on the above noise model about ϵ, derive the log-likelihood for the observed training data $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ and the objective function for computing the solution $\mathbf{w}$. Does the problem have a closed form solution like Least Square Regression?

Problem 2: Fitting Polynomial Functions (35 Points)

In class, we see that, a high-order polynomial, despite exhibiting superior fit to the training data, could result in diminished accuracy when applied to unseen testing data. Here, we try to observe it and try to improve it. Code to generate data and fit a model using sklearn is given in "p2.py" file. We measure the performance by the root mean square error (RMSE[1])

Each data point i originally has a feature $\mathbf{x}_i \in \mathbb{R}$, and a target $\mathbf{y}_i$. We first transfer it to polynomial features as $\phi(\mathbf{x}_i) = (\mathbf{x}_i, \mathbf{x}_i^2, \mathbf{x}_i^3, ..., \mathbf{x}_i^d)^\top$, where d is the degree. With a linear model $\mathbf{w} \in \mathbb{R}^d$ and w_0 to be a intercept term, the prediction is given by $\phi(\mathbf{x}_i)^\top \mathbf{w} + w_0$. Let $\Phi =$

[1]For a set of examples $(\mathbf{x}_i, y_i), i = 1, \ldots, n$, the root mean square error of a prediction function $f(\cdot)$ is computed by RMSE $= \sqrt{\sum_{i=1}^n (f(\mathbf{x}_i) - y_i)^2/n}$.

$(\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), ..., \phi(\mathbf{x}_n))^\top$ be a $\mathbb{R}^{n \times d}$ matrix, i.e., each row of Φ representing the polynomial features of one data point. We consider the following problem

$$\min_{\mathbf{w}} \frac{1}{2}\|\Phi\mathbf{w} + w_0 - \mathbf{y}\|_2^2 + \frac{\lambda}{2}\|\mathbf{w}\|_2^2,$$

1) Fix $n\_train = 10$ and $\lambda = 0$, vary degree in $[1, 5, 10, 20]$, i) Compare how training RMSE change. ii) Compare how testing RMSE change. Write your response to these questions in pdf (same below).

2) Fix $\lambda = 0$. Increase n_train to be 10000. Vary degree in $[1, 5, 10, 20]$. How are the training RMSE and testing RMSE compared to 1)?

3) Let $n\_train = 10$, $\lambda = 0.001$. Vary degree in $[1, 5, 10, 20]$. How are the training RMSE and testing RMSE compared to 1)?

Problem 3: Ridge Regression and Lasso (50 points)

In this problem, you are asked to learn regression models using Ridge regression and Lasso. The data set that we are going to use is the E2006-tfidf[2], which is included in the homework package.

The first column is the target output y, and the remaining columns are features in the form of (feature_index:feature_value). You can load the data by sklearn[3]. If we let $\mathbf{x} \in \mathbb{R}^d$ denote the feature vector, the prediction is given by $\mathbf{x}^\top\mathbf{w} + w_0$, where $\mathbf{w} \in \mathbb{R}^d$ contains the coefficients for all features and w_0 is a intercept term. Let $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n)^\top$ be a $\mathbb{R}^{n \times d}$ matrix, where each row of $\mathbf{X}$ denotes the features of one data point. The problem becomes

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{X}\mathbf{w} + w_0 - \mathbf{y}\|_2^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2,$$

which is the Lasso regression problem when the regularization term $\|\mathbf{w}\|^2 = \|\mathbf{w}\|_1^2$ and is the Ridge regression problem when the regularization term $\|\mathbf{w}\|^2 = \|\mathbf{w}\|_2^2$.

You can use the Python sklearn library for Lasso[4] and Ridge regression[5].

(1) Solution of Ridge Regression and Lasso: Set the value of the regularization parameter $\lambda = 0.1$, compute the optimal solution for Ridge regression and Lasso. Report the number of nonzero coefficient in the solution $\mathbf{w}$ for both Ridge regression and Lasso. (Note: When you use the sklearn Ridge, set alpha to be λ. When you use the sklearn Lasso, the value of alpha should be set to λ/n, where n is the number of training examples. Same for following questions. This is because an inconsistent implementation of sklearn.)

(2) Training and testing error with different values of λ: (i) For each value of λ in [1e-3, 1e-2, 0.1, 1, 100, 1e3, 1e4] run the Ridge regression and Lasso on training data to obtain a model $\mathbf{w}$ and then compute the root mean square error (RMSE[6]) on both the training and the testing data of the obtained model. (ii) Plot the error curves for RMSE on both the training data

[2] `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/regression.html`

[3] `https://scikit-learn.org/stable/datasets/loading_other_datasets.html`

[4] `http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html`

[5] `https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html`

[6] For a set of examples $(\mathbf{x}_i, y_i), i = 1, \ldots, n$, the root mean square error of a prediction function $f(\cdot)$ is computed by RMSE $= \sqrt{\sum_{i=1}^n (f(\mathbf{x}_i) - y_i)^2/n}$.

and the testing data vs different values of λ. You need to show the curves, and discuss your observations of the error curves, and report the best value of λ and the corresponding testing error. (iii) Plot the curve of number of nonzero elements in the solution $\mathbf{w}$ vs different values of λ. Discuss your observations.

(3) Cross-validation: Use the given training data and follow the 5-fold cross-validation procedure to select the best value of λ for both Ridge regression and Lasso. Then train the model on the whole training data using the selected λ and compute the root mean square error on the testing data. Report the best λ and the testing error for both Ridge regression and Lasso.

Problem 4: Regularization Penalizes Large Magnitudes of Parameters (Optional, 30 Bonus points)

Let $\phi(\mathbf{x}_i)$ be a column vector that represents the features for one data point. Define $\Phi = (\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_n))^\top \in \mathbb{R}^{n \times d}$. Consider the regularized least square problem

$$\min_{\mathbf{w}} \frac{1}{2}\|\Phi\mathbf{w} - \mathbf{y}\|_2^2 + \frac{\lambda}{2}\|\mathbf{w}\|_2^2.$$

The optimal solution $\mathbf{w}_*$ can be computed by

$$\mathbf{w}_* = (\lambda I + \Phi^\top \Phi)^{-1}\Phi^\top \mathbf{y}$$

In class, we have learned that when increasing the regularization parameter λ the magnitude of the optimal solution will decrease. In this homework, you are asked to pove this argument. To this end, you need to use SVD of Φ. Suppose $\Phi \in \mathbb{R}^{n \times d}$ $(n \geq d)$ has a singular value decomposition given by $\Phi = U\Sigma V^\top$, where $U \in \mathbb{R}^{n \times d}$ and $V \in \mathbb{R}^{d \times d}$ are orthonormal matrices satisfying $U^\top U = I_d$ and $V^\top V = I_d, VV^\top = I_d$, and $\Sigma = diag(\sigma_1, \cdots, \sigma_d)$ is a diagonal matrix with $\sigma_i \geq 0, i = 1, \ldots, d$.

1) Try to use SVD of Φ to simplify the following expression and transform it into a product of three matrices, where the middel matrix is a diagonal one:

$$(\lambda I_d + \Phi^\top \Phi)^{-1}\Phi^\top$$

using U, Σ and V, where I_d is an identity matrix of size $d \times d$ and $\lambda > 0$ is a constant.

2) Using the simplied expression from 2) to argue that the Euclidean norm of the optimal solution $\|\mathbf{w}_*\|_2$ will decrease as λ increases.

Hint: For any vector $\mathbf{u} \in \mathbb{R}^d$ if $V^\top V = I$ where $V \in \mathbb{R}^{d \times d}$ then $\|V\mathbf{u}\|_2 = \|\mathbf{u}\|_2$